Provided the functions in a library follow the appropriate run-time conventions (e.g., method of passing arguments), then these functions may be written in any other language.  
Trial-and-error/divide-and-conquer is needed: the programmer will try to remove some parts of the original test case and check if the problem still exists.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
Languages form an approximate spectrum from "low-level" to "high-level"; "low-level" languages are typically more machine-oriented and faster to execute, whereas "high-level" languages are more abstract and easier to use but execute less quickly.  
There are many approaches to the Software development process.  
However, Charles Babbage had already written his first program for the Analytical Engine in 1837.  
FORTRAN, the first widely used high-level language to have a functional implementation, came out in 1957, and many other languages were soon developed—in particular, COBOL aimed at commercial data processing, and Lisp for computer research.  
By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers.  
Transpiling on the other hand, takes the source-code from a high-level programming language and converts it into bytecode.  
  
 Readability is important because programmers spend the majority of their time reading, trying to understand, reusing and modifying existing source code, rather than writing new source code.  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.  
Provided the functions in a library follow the appropriate run-time conventions (e.g., method of passing arguments), then these functions may be written in any other language.  
However, Charles Babbage had already written his first program for the Analytical Engine in 1837.  
Many applications use a mix of several languages in their construction and use.